```
LLL                III IIIIII  BBBBBBBBBBBB  RRRRRRRRRRR   TTTTTTTTTTTTTTT  LLL
LLL                IIIIIIIIII   BBBBBBBBBBBB  RRRRRRRRRRR   TTTTTTTTTTTTTTT  LLL
LLL                IIIIIIIII    BBBBBBBBBBBB  RRRRRRRRRRR   TTTTTTTTTTTTTT   LLL
LLL                   III       BBB      BBB  RRR      RRR       TTT         LLL
LLL                   III       BBB      BBB  RRR      RRR       TTT         LLL
LLL                   III       BBB      BBB  RRR      RRR       TTT         LLL
LLL                   III       BBB      BBB  RRR      RRR       TTT         LLL
LLL                   III       BBB      BBB  RRR      RRR       TTT         LLL
LLL                   III       BBBBBBBBBBBB  RRRRRRRRRRR        TTT         LLL
LLL                   III       BBBBBBBBBBBB  RRRRRRRRRRR        TTT         LLL
LLL                   III       BBBBBBBBBBBB  RRRRRRRRRRR        TTT         LLL
LLL                   III       BBB      BBB  RRR   RRR          TTT         LLL
LLL                   III       BBB      BBB  RRR    RRR         TTT         LLL
LLL                   III       BBB      BBB  RRR     RRR        TTT         LLL
LLL                   III       BBB      BBB  RRR      RRR       TTT         LLL
LLL                   III       BBB      BBB  RRR      RRR       TTT         LLL
LLLLLLLLLLLLLLL    IIIIIIIII    BBBBBBBBBBBB  RRR      RRR       TTT         LLLLLLLLLLLLLL
LLLLLLLLLLLLLLL    IIIIIIIII    BBBBBBBBBBBB  RRR      RRR       TTT         LLLLLLLLLLLLLL
LLLLLLLLLLLLLLL    IIIIIIIII    BBBBBBBBBBBB  RRR      RRR       TTT         LLLLLLLLLLLLLL
```

```
LL          IIIIII  BBBBBBBB   RRRRRRRR  EEEEEEEEEE  MM      MM  QQQQQQ    TTTTTTTTTT  IIIIII
LL          IIIIII  BBBBBBBB   RRRRRRRR  EEEEEEEEEE  MM      MM  QQQQQQ    TTTTTTTTTT  IIIIII
LL            II    BB     BB  RR     RR EE          MMMM  MMMM  QQ    QQ      TT          II
LL            II    BB     BB  RR     RR EE          MMMM  MMMM  QQ    QQ      TT          II
LL            II    BB     BB  RR     RR EE          MM  MM  MM  QQ    QQ      TT          II
LL            II    BB     BB  RR     RR EE          MM  MM  MM  QQ    QQ      TT          II
LL            II    BBBBBBBB   RRRRRRRR  EEEEEEE     MM      MM  QQ    QQ      TT          II
LL            II    BBBBBBBB   RRRRRRRR  EEEEEEE     MM      MM  QQ    QQ      TT          II
LL            II    BB     BB  RR  RR    EE          MM      MM  QQ QQ QQ      TT          II
LL            II    BB     BB  RR   RR   EE          MM      MM  QQ QQ QQ      TT          II
LL            II    BB     BB  RR    RR  EE          MM      MM  QQ   QQ       TT          II
LL            II    BB     BB  RR    RR  EE          MM      MM  QQ   QQ       TT          II
LLLLLLLLLL  IIIIII  BBBBBBBB   RR     RR EEEEEEEEEE  MM      MM  QQQQ QQ       TT        IIIIII
LLLLLLLLLL  IIIIII  BBBBBBBB   RR     RR EEEEEEEEEE  MM      MM  QQQQ QQ       TT        IIIIII


LL          IIIIII   SSSSSSSS
LL          IIIIII   SSSSSSSS
LL            II    SS
LL            II    SS
LL            II    SS
LL            II     SSSSSS
LL            II     SSSSSS
LL            II          SS
LL            II          SS
LL            II          SS
LL            II          SS
LLLLLLLLLL  IIIIII   SSSSSSSS
LLLLLLLLLL  IIIIII   SSSSSSSS
```

LIB$REMQTI
1-002

L 6
- Remove Entry from Queue at Tail, Inter 16-SEP-1984 00:17:48 VAX/VMS Macro V04-00   Page 1
                                         6-SEP-1984 11:10:15 [LIBRTL.SRC]LIBREMQTI.MAR;1   (1)

```
0000    1          .TITLE  LIB$REMQTI - Remove Entry from Queue at Tail, Interlocked
0000    2          .IDENT  /1-002/                 ; File: LIBREMQTI.MAR Edit: DGP1002
0000    3
0000    4 ;
0000    5 ;********************************************************************************
0000    6 ;*                                                                              *
0000    7 ;* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                      *
0000    8 ;* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                       *
0000    9 ;* ALL RIGHTS RESERVED.                                                         *
0000   10 ;*                                                                              *
0000   11 ;* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED        *
0000   12 ;* ONLY IN ACCORDANCE WITH THE  TERMS  OF  SUCH  LICENSE  AND WITH THE          *
0000   13 ;* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER        *
0000   14 ;* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY        *
0000   15 ;* OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY        *
0000   16 ;* TRANSFERRED.                                                                 *
0000   17 ;*                                                                              *
0000   18 ;* THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE        *
0000   19 ;* AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT        *
0000   20 ;* CORPORATION.                                                                 *
0000   21 ;*                                                                              *
0000   22 ;* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS        *
0000   23 ;* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                       *
0000   24 ;*                                                                              *
0000   25 ;*                                                                              *
0000   26 ;********************************************************************************
0000   27 ;
0000   28
0000   29 ;++
0000   30 ; FACILITY: General Utility Library
0000   31 ;
0000   32 ; ABSTRACT:
0000   33 ;
0000   34 ;       One of four procedures which give higher level languages access
0000   35 ;       to the interlocked, self-relative queue instructions on the
0000   36 ;       VAX-11/780 and all future machines. This library procedure permits
0000   37 ;       the high level language user to have access to the REMQTI instruction.
0000   38 ;
0000   39 ; ENVIRONMENT: User Mode, AST Reentrant
0000   40 ;
0000   41 ;--
0000   42 ; AUTHOR: R. E. Johnston, CREATION DATE: 03-Dec-79
0000   43 ;
0000   44 ; MODIFIED BY:
0000   45 ;
0000   46 ; 1-001 - Original. REJ 03-Dec-79
0000   47 ; 1-002 - Retry count is off by one.  DGP 14-Aug-1981
0000   48 ;--
```

LIB$REMQTI
1-002

M 6
- Remove Entry from Queue at Tail, Inter 16-SEP-1984 00:17:48   VAX/VMS Macro V04-00   Page   2
DECLARATIONS   6-SEP-1984 11:10:15   [LIBRTL.SRC]LIBREMQTI.MAR;1   (2)

LIB$
1-00

```
                      0000      50          .SBTTL  DECLARATIONS
                      0000      51 ;
                      0000      52 ; INCLUDE FILES:
                      0000      53 ;
                      0000      54
                      0000      55
                      0000      56 ; EXTERNAL DECLARATIONS:
                      0000      57 ;
                      0000      58          .DSABL  GBL                    ; Disable automatic generation
                      0000      59                                         ; of .EXTRN
                      0000      60          .EXTRN  SS$_NORMAL             ; Normal successful completion
                      0000      61          .EXTRN  LIB$_SECINTFAI         ; Secondary Interlock still
                      0000      62                                         ; locked after retry-cnt retrys
                      0000      63          .EXTRN  LIB$_ONEENTQUE         ; Successful Completion but
                      0000      64                                         ; the queue is now empty
                      0000      65          .EXTRN  LIB$_QUEWASEMP         ; Queue was empty
                      0000      66                                         ; Queue is not modified
                      0000      67                                         ; Procedure is not successful
                      0000      68 ; MACROS:
                      0000      69 ; MACROS:
                      0000      70 ;
                      0000      71
                      0000      72
                      0000      73 ; EQUATED SYMBOLS:
                      0000      74 ;
            0000000A  0000      75          DEF_RETRY_CNT = 10             ; Default retry count for
                      0000      76                                         ; Secondary Interlock fails
                      0000      77 ;
                      0000      78 ; OWN STORAGE:
                      0000      79 ;
                      0000      80
                      0000      81 ;
                      0000      82 ; PSECT DECLARATIONS:
                      0000      83 ;
            00000000            84          .PSECT _LIB$CODE PIC, SHR, LONG, EXE, NOWRT
                      0000      85
                      0000      86
                      0000      87
```

LIB$REMQTI
1-002

N 6

- Remove Entry from Queue at Tail, Inter 16-SEP-1984 00:17:48   VAX/VMS Macro V04-00   Page 3
LIB$REMQTI - Remove Entry from Queue Tai 6-SEP-1984 11:10:15 [LIBRTL.SRC]LIBREMQTI.MAR;1   (3)

```
                           0000   89              .SBTTL LIB$REMQTI - Remove Entry from Queue Tail
                           0000   90   ;++
                           0000   91   ; FUNCTIONAL DESCRIPTION:
                           0000   92   ;
                           0000   93   ;
                           0000   94   ;     One of four procedures which give higher level languages access
                           0000   95   ;     to the interlocked, self-relative queue instructions on the
                           0000   96   ;     VAX-11/780 and all future machines. This library procedure permits
                           0000   97   ;     the high level language user to have access to the REMQTI instruction.
                           0000   98   ;     With this procedure the user may remove a queue entry from the tail of
                           0000   99   ;     a user specified queue.
                           0000  100   ;
                           0000  101   ;     If the entry is successfully removed from the tail of the queue and the
                           0000  102   ;     queue now contains one or more entries, a successful completion status
                           0000  103   ;     is returned. If the entry is removed from the tail of the queue and no
                           0000  104   ;     other entries are now in the queue, the execution is successful but a
                           0000  105   ;     unique status value is returned indicating that the queue now contains
                           0000  106   ;     no entries (LIB$_ONEENTQUE).
                           0000  107   ;
                           0000  108   ;     These queue instructions are synchronized across all processors
                           0000  109   ;     through the use of a secondary interlock. The user may specify a
                           0000  110   ;     secondary interlock retry count. (The default retry count is 10.)
                           0000  111   ;     If the secondary interlock remains locked through retry-count retrys,
                           0000  112   ;     a secondary interlock status is returned to the user (LIB$_SECINTFAI)
                           0000  113   ;     and the entry is NOT successfully removed from the tail of the queue.
                           0000  114   ;
                           0000  115   ;     If an attempt is made to remove an entry from a queue which is already
                           0000  116   ;     empty, a unique unsuccessful completion status is returned to the
                           0000  117   ;     user (LIB$_QUEWASEMP).
                           0000  118   ;
                           0000  119   ; CALLING SEQUENCE:
                           0000  120   ;
                           0000  121   ;   ret-status.wlc.v = LIB$REMQTI (header.mq.r, addr.wl.r[, retry-cnt.rlu.r])
                           0000  122   ;
                           0000  123   ;
                           0000  124   ; INPUT PARAMETERS:
                           0000  125   ;
              00000004     0000  126   ;     HEADER = 4                              ; Address of queue header
              00000008     0000  127   ;     ADDR = 8                                ; Address where queue entry address
                           0000  128   ;                                             ; is to be returned to user
              0000000C     0000  129   ;     RETRY_CNT = 12                          ; Address of retry count
                           0000  130   ;
                           0000  131   ; IMPLICIT INPUTS:
                           0000  132   ;
                           0000  133   ;     NONE
                           0000  134   ;
                           0000  135   ; OUTPUT PARAMETERS:
                           0000  136   ;
                           0000  137   ;     NONE
                           0000  138   ;
                           0000  139   ; IMPLICIT OUTPUTS:
                           0000  140   ;
                           0000  141   ;     NONE
                           0000  142   ;
                           0000  143   ; FUNCTION VALUE:
                           0000  144   ;
                           0000  145   ;     SS$_NORMAL      - Entry removed from tail of queue, queue still contains
```

LIB$REMQTI
1-002

B 7
- Remove Entry from Queue at Tail, Inter 16-SEP-1984 00:17:48 VAX/VMS Macro V04-00 Page 4
LIB$REMQTI - Remove Entry from Queue Tai 6-SEP-1984 11:10:15 [LIBRTL.SRC]LIBREMQTI.MAR;1 (3)

```
                    0000  146 ;                              one or more entries
                    0000  147 ;          LIB$_ONEENTQUE - Successful completion of instruction (REMQTI).
                    0000  148 ;                           Entry removed from tail of queue, but queue is now
                    0000  149 ;                           empty.
                    0000  150 ;          LIB$_SECINTFAI - Secondary Interlock failed, queue is not modified.
                    0000  151 ;          LIB$_QUEWASEMP - Unsuccessful completion of instruction (REMQTI).
                    0000  152 ;                           The queue was empty before the instruction was
                    0000  153 ;                           executed.
                    0000  154 ;
                    0000  155 ; SIDE EFFECTS:
                    0000  156 ;
                    0000  157 ;          SS$_ROPRAND - reserved operand fault for:
                    0000  158 ;                        1.) either the entry or the header is at an address
                    0000  159 ;                            that is not quad word aligned.
                    0000  160 ;                        2.) address of header equals address of entry.
                    0000  161 ;
                    0000  162 ;--
                    0000  163
               0000 0000  164          .ENTRY  LIB$REMQTI , ^M< >       ; Entry point
                    0002  165
        50    0A  D0 0002  166          MOVL    #DEF_RETRY_CNT, R0       ; R0 = Default retry count of 10
        03    6C  91 0005  167          CMPB    (AP), #<RETRY_CNT/4>     ; Check for optional retry cnt operand
              04  1F 0008  168          BLSSU   20$                     ; Branch if default count to be used
        50  0C BC  D0 000A  169          MOVL    @RETRY_CNT(AP), R0       ; R0 = User specified retry count
                    000E  170 20$:
    08 BC  04 BC  5F 000E  171          REMQTI  @HEADER(AP), @ADDR(AP)   ; Do the instruction (REMQTI)
              14  1F 0013  172          BCS     40$                     ; Branch if C = 1
                    0015  173                                          ; (Secondary Interlock fail)
              08  13 0015  174          BEQL    30$                     ; Branch if Z = 1
                    0017  175                                          ; (Queue is now empty)
    50  00000000'8F  D0 0017  176          MOVL    #SS$_NORMAL, R0         ; Normal status - Entry removed from
                    001E  177                                          ; tail of queue and one or more entries
                    001E  178                                          ; are still in queue
              04  001E  179          RET                             ; Successful return to user
                    001F  180 30$:
              13  1D 001F  181          BVS     50$                     ; Branch if V = 1
                    0021  182                                          ; (There was nothing to remove)
    50  00000000'8F  D0 0021  183          MOVL    #LIB$_ONEENTQUE, R0     ; Assume the queue is just now empty
                    0028  184                                          ; Entry successfully removed from queue
              04  0028  185          RET                             ; Successful return to user
                    0029  186 40$:
        E2 50  F4 0029  187          SOBGEQ  R0, 20$                 ; Loop until retry count is exhausted
    50  00000000'8F  D0 002C  188          MOVL    #LIB$_SECINTFAI, R0     ; Retry count is exhausted
                    0033  189                                          ; Secondary Interlock fail status
              04  0033  190          RET                             ; Unsuccessful return to user
                    0034  191 50$:
    50  00000000'8F  D0 0034  192          MOVL    #LIB$_QUEWASEMP, R0     ; Queue was already empty before
                    003B  193                                          ; this queue instruction was executed
              04  003B  194          RET                             ; Unsuccessful return to user
                    003C  195          .END
```

C 7

LIB$REMQTI   - Remove Entry from Queue at Tail, Inter 16-SEP-1984 00:17:48   VAX/VMS Macro V04-00   Page   5
Symbol table   6-SEP-1984 11:10:15   [LIBRTL.SRC]LIBREMQTI.MAR;1   (3)

```
ADDR            = 00000008
DEF_RETRY_CNT   = 0000000A
HEADER          = 00000004
LIB$REMQTI        00000000 RG   01
LIB$_ONEENTQUE    ********  X    00
LIB$_QUEWASEMP    ********  X    00
LIB$_SECINTFAI    ********  X    00
RETRY_CNT       = 0000000C
SS$_NORMAL        ********  X    00
```

```
+-----------------+
! Psect synopsis !
+-----------------+
```

PSECT name	Allocation		PSECT No.	Attributes										
. ABS .	00000000 (	0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE
_LIB$CODE	0000003C (	60.)	01 (1.)	PIC	USR	CON	REL	LCL SHR	EXE	RD	NOWRT	NOVEC	LONG

```
+-----------------------+
! Performance indicators !
+-----------------------+
```

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.05	00:00:03.06
Command processing	107	00:00:00.31	00:00:02.17
Pass 1	69	00:00:00.32	00:00:02.86
Symbol table sort	0	00:00:00.01	00:00:00.01
Pass 2	53	00:00:00.22	00:00:03.01
Symbol table output	2	00:00:00.01	00:00:00.02
Psect synopsis output	2	00:00:00.01	00:00:00.01
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	265	00:00:00.95	00:00:11.14

The working set limit was 900 pages.
2041 bytes (4 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 9 non-local and 4 local symbols.
195 source lines were read in Pass 1, producing 11 object records in Pass 2.
0 pages of virtual memory were used to define 0 macros.

```
+-----------------------------+
! Macro library statistics !
+-----------------------------+
```

Macro library name	Macros defined
_255DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS$:LIBREMQTI/OBJ=OBJ$:LIBREMQTI MSRC$:LIBREMQTI/UPDATE=(ENH$:LIBREMQTI)

LIBPOLYG
LIS

LIBREMQHI
LIS

LIBSIGSTO
LIS

LIBRENAME
LIS

LIBSCANC
LIS

LIBRDOBJ
LIS

LIBRUNPRO
LIS

LIBSIGNAL
LIS

LIBPUTOUT
LIS

LIBREMQTI
LIS

LIBSIGRET
LIS

LIBSIMTRA
LIS

LIBSCOPY
LIS

LIBPOLYH
LIS

LIBREVERT
LIS